

# I217: Functional Programming

## 11. Program Verification – Natural Numbers

Kazuhiro Ogata

### Roadmap

- Natural Numbers a la Peano
- Associativity of  $\_ + \_$
- Commutativity of  $\_ + \_$
- Associativity of  $\_ * \_$
- Commutativity of  $\_ * \_$
- Correctness of a Tail Recursive Factorial

## Natural Numbers a la Peano

Natural numbers have been formalized by Giuseppe Peano (1858 – 1932), an Italian mathematician.

Inductively defined as follows:

(1) 0 is a natural number.

(2) If  $n$  is a natural number, then the successor of  $n$ , denoted  $s(n)$ , is also a natural number.

0	$s(0)$	$s(s(0))$	$s(s(s(0)))$	$s(s(s(s(0))))$
0	1	2	3	4

## Natural Numbers a la Peano

Natural numbers can be specified in CafeOBJ as follows:

```

mod! PNAT1 {
  [PNat]
  op 0 : -> PNat {constr} .
  op s : PNat -> PNat {constr} .
  ...
}

```

Terms 0,  $s(0)$ ,  $s(s(0))$ ,  $s(s(s(0)))$ ,  $s(s(s(s(0))))$  denote 0, 1, 2, 3, 4.

## Natural Numbers a la Peano

For every sort  $S$ , the following operator and equations are prepared in the built-in module  $EQL$  that is imported by

**BOOL:**     **op**  $\_ = \_ : S \ S \rightarrow \text{Bool} \ \{\text{comm}\} \ .$   
               **eq**  $(X = X) = \text{true} \ .$   
               **eq**  $(\text{true} = \text{false}) = \text{false} \ .$

where  $X$  is a variable of  $S$ .

We declare the following equations in  $PNAT1$  for  $PNat$ :

**eq**  $(0 = s(Y)) = \text{false} \ .$   
**eq**  $(s(X) = s(Y)) = (X = Y) \ .$

where  $X$  and  $Y$  are variables of  $PNat$ .

Let  $X$ ,  $Y$  and  $Z$  be variables of  $PNat$  in the rest of the slides.

## Natural Numbers a la Peano

Addition of natural numbers is defined as follows:

**op**  $\_ + \_ : PNat \ PNat \rightarrow PNat \ .$   
**eq**  $0 + Y = Y \ .$                                 **-- (+1)**  
**eq**  $s(X) + Y = s(X + Y) \ .$                 **-- (+2)**

$s(s(s(0))) + s(s(s(s(0))))$   
 $\rightarrow s(\underline{s(s(0)) + s(s(s(s(0))))})$             **by (+2)**  
 $\rightarrow s(s(\underline{s(0) + s(s(s(s(0))))}))$             **by (+2)**  
 $\rightarrow s(s(s(\underline{0 + s(s(s(s(0))))}))$             **by (+2)**  
 $\rightarrow s(s(s(s(s(s(s(0)))))))$                 **by (+1)**

## Natural Numbers a la Peano

Multiplication of natural numbers is defined as follows:

```

op _ * _ : PNat PNat -> PNat .
eq 0 * Y = 0 .                -- (*1)
eq s(X) * Y = (X * Y) + Y .  -- (*2)

```

```

s(s(0)) * s(s(0))
→ (s(0) * s(s(0))) + s(s(0))          by (*2)
→ ((0 * s(s(0))) + s(s(0))) + s(s(0))  by (*2)
→ (0 + s(s(0))) + s(s(0))              by (*1)
→ s(s(0)) + s(s(0))                    by (+1)
→ s(s(0) + s(s(0)))                     by (+2)
→ s(s(0 + s(s(0))))                     by (+2)
→ s(s(s(0)))                             by (+1)

```

## Natural Numbers a la Peano

Factorial function can be defined as follows:

```

op fact1 : PNat -> PNat .
eq fact1(0) = s(0) .          -- (f1-1)
eq fact1(s(X)) = s(X) * fact1(X) .  -- (f1-2)

```

Another implementation (a tail recursive version) of Factorial function is as follows:

```

op fact2 : PNat -> PNat .
op sfact2 : PNat PNat -> PNat .
eq fact2(X) = sfact2(X,s(0)) .      -- (f2)
eq sfact2(0,Y) = Y .                -- (sf2-1)
eq sfact2(s(X),Y) = sfact2(X,s(X) * Y) .  -- (sf2-2)

```

## Natural Numbers a la Peano

Let  $l(X)$  and  $r(X)$  be terms of a same sort (say  $\text{PNat}$ ) in which a variable  $X$  of  $\text{PNat}$  occurs. Then, the following (A) and (B) are equivalent:

$$(A) (\forall X:\text{PNat}) l(X) = r(X)$$

$$(B) \text{ I. } l(0) = r(0)$$

II. If  $l(x) = r(x)$ , then  $l(s(x)) = r(s(x))$ , where  $x$  is a fresh constant of  $\text{PNat}$ .

It suffices to prove (B) so as to prove (A). This is called proof by *structural induction* on natural number  $X$ . I is called the *base case*, II is called the *induction case*, and  $l(x) = r(x)$  is called the *induction hypothesis*.

## Associativity of $\_+\_$

$(X + Y) + Z$  equals  $X + (Y + Z)$  for all natural numbers  $X, Y, Z$ . This is what we will prove by structural induction on  $X$ .

Theorem 1 [Associativity of  $\_+\_$  (assoc+)]

$$(\forall X:\text{PNat}) ((X + Y) + Z = X + (Y + Z))$$

In the rest of the slides, the universal quantifiers, such as  $(\forall X:\text{PNat})$ , are omitted, and the above formula is written as follows:

$$(X + Y) + Z = X + (Y + Z)$$

## Associativity of $_{+}$

Proof of Theorem 1 By structural induction on  $X$ .

Let  $x, y, z$  be fresh constants of  $\text{PNat}$ .

I. Base case

What to show is  $(0 + y) + z = 0 + (y + z)$ .

$(0 + y) + z \rightarrow y + z$  by (+1)       $0 + (y + z) \rightarrow y + z$  by (+1)

II. Induction case

What to show is  $(s(x) + y) + z = s(x) + (y + z)$

assuming the induction hypothesis  $(x + Y) + Z = x + (Y + Z)$  -- (IH)

$(s(x) + y) + z \rightarrow s(x + y) + z$  by (+2)  
 $\rightarrow s((x + y) + z)$  by (+2)  
 $\rightarrow s(x + (y + z))$  by (IH)

$s(x) + (y + z) \rightarrow s(x + (y + z))$  by (+2)

End of Proof of Theorem 1

## Associativity of $_{+}$

Part of the proof is written in CafeOBJ. Proofs written in CafeOBJ are called *proof scores*.

Proof of Theorem 1 By structural induction on  $X$ .

I. Base case

```
open PNAT1 .
-- fresh constants
ops y z : -> PNat .
-- check
red (0 + y) + z = 0 + (y + z) .
close
```

II. Induction case

```
open PNAT1 .
-- fresh constants
ops x y z : -> PNat .
-- IH
eq (x + Y) + Z = x + (Y + Z) .
-- check
red (s(x) + y) + z = s(x) + (y + z) .
close
```

End of Proof of Theorem 1      Please see the appendices for the proof score.

## Commutativity of $\_+\_$

Theorem 2 [Commutativity of  $\_+\_$  (comm+)]  $X + Y = Y + X$

Proof of Theorem 2 By structural induction on  $X$ .

Let  $x, y$  be fresh constants of  $\mathbf{PNat}$ .

I. Base case

What to show is  $0 + y = y + 0$ .

$0 + y \rightarrow y$  by (+1)

Since  $y + 0$  cannot be rewritten, we need a lemma that makes it possible to conclude that  $y + 0$  equals  $0$ . We conjecture the following lemma:

$$X + 0 = X \quad \text{-- (rz+)}$$

$y + 0 \rightarrow y$  by (rz+)

(The lemma will be proved later.)

## Commutativity of $\_+\_$

II. Induction case

What to show is  $s(x) + y = y + s(x)$

assuming the induction hypothesis  $x + Y = Y + x$  -- (IH)

$s(x) + y \rightarrow s(x + y)$  by (+2)  
 $\rightarrow s(y + x)$  by (IH)

$y + s(x)$  cannot be rewritten, and then we need a lemma that makes it possible to conclude that  $y + s(x)$  equals  $s(y + x)$ . We conjecture the following lemma:

$$X + s(Y) = s(X + Y) \quad \text{-- (rs+)}$$

$y + s(x) \rightarrow s(y + x)$  by (rs+)

End of Proof of Theorem 2

(The lemma will be proved later.)

## Commutativity of $\_+ \_$

Lemma 1 [Right zero of  $\_+ \_$  (rz+)]  $X + 0 = X$

Proof of Lemma 1 By structural induction on  $X$ .

Let  $x$  be fresh constants of  $\mathbf{PNat}$ .

I. Base case

What to show is  $0 + 0 = 0$ .

$0 + 0 \rightarrow 0$  by (+1)

II. Induction case

What to show is  $s(x) + 0 = s(x)$

assuming the induction hypothesis  $x + 0 = x$  -- (IH)

$s(x) + 0 \rightarrow s(x + 0)$  by (+2)  
 $\rightarrow s(x)$  by (IH)

End of Proof of Lemma 1

## Commutativity of $\_+ \_$

Lemma 2 [Right successor of  $\_+ \_$  (rs+)]  $X + s(Y) = s(X + Y)$

Proof of Lemma 2 By structural induction on  $X$ .

Let  $x, y$  be fresh constants of  $\mathbf{PNat}$ .

I. Base case

What to show is  $0 + s(y) = s(0 + y)$ .

$0 + s(y) \rightarrow s(y)$  by (+1)       $s(0 + y) \rightarrow s(y)$  by (+1)

II. Induction case

What to show is  $s(x) + s(y) = s(s(x) + y)$

assuming the induction hypothesis  $x + s(Y) = s(x + Y)$  -- (IH)

$s(x) + s(y) \rightarrow s(x + s(y))$  by (+2)       $s(s(x) + y) \rightarrow s(s(x + y))$  by (+2)  
 $\rightarrow s(s(x + y))$  by (IH)

End of Proof of Lemma 2



## Commutativity of $\_+\_$

Lemma 1 [Right zero of  $\_+\_$  (rz+)]  $X + 0 = X$

Proof of Lemma 1 By structural induction on  $X$ .

I. Base case

```
open PNAT1 .
-- check
red 0 + 0 = 0 .
close
```

II. Induction case

```
open PNAT1 .
-- fresh constants
op x : -> PNat .
-- IH
eq x + 0 = x .
-- check
red s(x) + 0 = s(x) .
close
```

End of Proof of Lemma 1

## Commutativity of $\_+\_$

Lemma 2 [Right successor of  $\_+\_$  (rs+)]  $X + s(Y) = s(X + Y)$

Proof of Lemma 2 By structural induction on  $X$ .

I. Base case

```
open PNAT1 .
-- fresh constants
op y : -> PNat .
-- check
red 0 + s(y) = s(0 + y) .
close
```

II. Induction case

```
open PNAT1 .
-- fresh constants
ops x y : -> PNat .
-- IH
eq x + s(Y) = s(x + Y) .
-- check
red s(x) + s(y) = s(s(x) + y) .
close
```

End of Proof of Lemma 2

## Commutativity of $\_+\_$

**Theorem 2** [Commutativity of  $\_+\_$  (comm+)]  $X + Y = Y + X$

**Proof of Theorem 2** By structural induction on  $X$ .

I. Base case

```
open PNAT1 .
-- fresh constants
op y : -> PNat .
-- lemmas
eq X + 0 = X . -- (rz+)
-- check
red 0 + y = y + 0 .
close
```

II. Induction case

```
open PNAT1 .
-- fresh constants
ops x y : -> PNat .
-- lemmas
eq X + s(Y) = s(X + Y) . -- (rs+)
-- IH
eq x + Y = Y + x .
-- check
red s(x) + y = y + s(x) .
close
```

**End of Proof of Theorem 2**

## Associativity of $\_*\_$

**Theorem 3** [Associativity of  $\_*\_$  (assoc\*)]  $(X * Y) * Z = X * (Y * Z)$

**Proof of Theorem 3** By structural induction on  $X$ .

Let  $x, y, z$  be fresh constants of  $\text{PNat}$ .

I. Base case

What to show is  $(0 * y) * z = 0 * (y * z)$ .

$(0 * y) * z \rightarrow 0 * z$	by (*)
$\rightarrow 0$	by (*)

$0 * (y * z) \rightarrow 0$	by (*)
-----------------------------	--------

## Associativity of $_*$

### II. Induction case

What to show is  $(s(x) * y) * z = s(x) * (y * z)$

assuming the induction hypothesis  $(x * Y) * Z = x * (Y * Z) \quad \text{-- (IH)}$

$$(s(x) * y) * z \rightarrow ((x * y) + y) * z \quad \text{by } (*2)$$

$$s(x) * (y * z) \rightarrow (x * (y * z)) + (y * z) \quad \text{by } (*2)$$

We cannot make the two terms rewritten to a same term and then need a lemma to do so. It seems sufficient to conjecture the following one:

$$(X + Y) * Z = (X * Z) + (Y * Z) \quad \text{-- (d*o+)}$$

$$\begin{aligned} ((x * y) + y) * z &\rightarrow ((x * y) * z) + (y * z) && \text{by (d*o+)} \\ &\rightarrow (x * (y * z)) + (y * z) && \text{by (IH)} \end{aligned}$$

End of Proof of Theorem 3

## Associativity of $_*$

Lemma 3 [Distributive law of  $_*$  over  $_+$  (d\*o+)]

$$(X + Y) * Z = (X * Z) + (Y * Z)$$

Proof of Lemma 3 By structural induction on  $X$ .

Let  $x, y, z$  be fresh constants of  $\mathbf{PNat}$ .

### I. Base case

What to show is  $(0 + y) * z = (0 * z) + (y * z)$ .

$$(0 + y) * z \rightarrow y * z \quad \text{by } (+1)$$

$$\begin{aligned} (0 * z) + (y * z) &\rightarrow 0 + (y * z) && \text{by } (*1) \\ &\rightarrow y * z && \text{by } (+1) \end{aligned}$$

## Associativity of $\_*$

### II. Induction case

What to show is  $(s(x) + y) * z = (s(x) * z) + (y * z)$

assuming the induction hypothesis

$(x + Y) * Z = (x * Z) + (Y * Z) \quad \text{-- (IH)}$

$$\begin{aligned}
 (s(x) + y) * z &\rightarrow s(x + y) * z && \text{by (+2)} \\
 &\rightarrow ((x + y) * z) + z && \text{by (*2)} \\
 &\rightarrow ((x * z) + (y * z)) + z && \text{by (IH)} \\
 &\rightarrow (x * z) + ((y * z) + z) && \text{by (assoc+)} \\
 &\rightarrow (x * z) + (z + (y * z)) && \text{by (comm+)}
 \end{aligned}$$

$$\begin{aligned}
 (s(x) * z) + (y * z) &\rightarrow ((x * z) + z) + (y * z) && \text{by (*2)} \\
 &\rightarrow (x * z) + (z + (y * z)) && \text{by (assoc+)}
 \end{aligned}$$

End of Proof of Lemma 3

## Associativity of $\_*$

Lemma 3 [Distributive law of  $\_*$  over  $\_+$  (d\*o+)]

$(X + Y) * Z = (X * Z) + (Y * Z)$

Proof of Lemma 3 By structural induction on  $X$ .

### I. Base case

```

open PNAT2 .
-- fresh constants
ops y z : -> PNat .
-- check
red (0 + y) * z = (0 * z) + (y * z) .
close

```

End of Proof of Lemma 3

### II. Induction case

```

open PNAT2 .
-- fresh constants
ops x y z : -> PNat .
-- IH
eq (x + Y) * Z = (x * Z) + (Y * Z) .
-- check
red (s(x) + y) * z = (s(x) * z) + (y * z) .
close

```

Note that PNAT2 is PNAT1 in which  $\_+$  is declared as follows:

```

op  $\_+$  : PNat PNat -> PNat {assoc comm} .

```

## Associativity of $_*$

Theorem 3 [Associativity of  $_*$  (assoc\*)]  $(X * Y) * Z = X * (Y * Z)$

Proof of Theorem 3 By structural induction on  $X$ .

I. Base case

```
open PNAT2 .
-- fresh constants
ops y z : -> PNat .
-- check
red (0 * y) * z = 0 * (y * z) .
close
```

II. Induction case

```
open PNAT2 .
-- fresh constants
ops x y z : -> PNat .
-- lemmas
eq (X + Y) * Z = (X * Z) + (Y * Z) . -- (d*o+)
-- IH
eq (x * Y) * Z = x * (Y * Z) .
-- check
red (s(x) * y) * z = s(x) * (y * z) .
close
```

End of Proof of Theorem 3

## Commutativity of $_*$

Theorem 4 [Commutativity of  $_*$  (assoc\*)]  $X * Y = Y * X$

Proof of Theorem 4 By structural induction on  $X$ .

Let  $x, y$  be fresh constants of  $PNat$ .

I. Base case

What to show is  $0 * y = y * 0$ .

$0 * y \rightarrow 0$  by (\*1)

We need the following lemma to make progress in the proof:

$$X * 0 = 0 \quad \text{-- (rz*)}$$

$y * 0 \rightarrow 0$  by (rz\*)

## Commutativity of $_ * _$

### II. Induction case

What to show is  $s(x) * y = y * s(x)$

assuming the induction hypothesis  $x * Y = Y * x \quad \text{-- (IH)}$

$$\begin{aligned} s(x) * y &\rightarrow (x * y) + y && \text{by (*2)} \\ &\rightarrow (y * x) + y && \text{by (IH)} \end{aligned}$$

We need the following lemma to make progress in the proof:

$$X * s(Y) = (X * Y) + X \quad \text{-- (rs*)}$$

$$y * s(x) \rightarrow (y * x) + y \quad \text{by (rs*)}$$

End of Proof of Theorem 4

## Commutativity of $_ * _$

Lemma 4 [Right zero of  $_ * _$  (rz\*)]  $X * 0 = 0$

Proof of Lemma 4 By structural induction on  $X$ .

Let  $x$  be a fresh constants of  $\mathbf{PNat}$ .

### I. Base case

What to show is  $0 * 0 = 0$ .

$$0 * 0 \rightarrow 0 \quad \text{by (*1)}$$

### II. Induction case

What to show is  $s(x) * 0 = 0$

assuming the induction hypothesis  $x * 0 = 0 \quad \text{-- (IH)}$

$$\begin{aligned} s(x) * 0 &\rightarrow (x * 0) + 0 && \text{by (*2)} \\ &\rightarrow 0 + 0 && \text{by (IH)} \\ &\rightarrow 0 && \text{by (+1)} \end{aligned}$$

End of Proof of Lemma 4

## Commutativity of $\_*$

**Lemma 5** [Right successor of  $\_*$  ( $rs^*$ )]  $X * s(Y) = (X * Y) + X$

**Proof of Lemma 5** By structural induction on  $X$ .

Let  $x, y$  be a fresh constants of  $\mathbf{PNat}$ .

I. Base case

What to show is  $0 * s(y) = (0 * y) + 0$ .

$$0 * s(y) \rightarrow 0 \quad \text{by } (*1)$$

$$\begin{aligned} (0 * y) + 0 &\rightarrow 0 + 0 && \text{by } (*1) \\ &\rightarrow 0 && \text{by } (+1) \end{aligned}$$

## Commutativity of $\_*$

II. Induction case

What to show is  $s(x) * s(y) = (s(x) * y) + s(x)$

assuming the induction hypothesis  $x * s(Y) = (x * Y) + x$  -- (IH)

$$\begin{aligned} s(x) * s(y) &\rightarrow (x * s(y)) + s(y) && \text{by } (*2) \\ &\rightarrow ((x * y) + x) + s(y) && \text{by (IH)} \\ &\rightarrow s(y) + ((x * y) + x) && \text{by (assoc+)} \\ &\rightarrow s(y + ((x * y) + x)) && \text{by } (+2) \end{aligned}$$

$$\begin{aligned} (s(x) * y) + s(x) &\rightarrow ((x * y) + y) + s(x) && \text{by } (*2) \\ &\rightarrow s(x) + ((x * y) + y) && \text{by (comm+)} \\ &\rightarrow s(x + ((x * y) + y)) && \text{by } (+2) \\ &\rightarrow s(((x * y) + y) + x) && \text{by (comm+)} \\ &\rightarrow s((y + (x * y)) + x) && \text{by (comm+)} \\ &\rightarrow s(y + ((x * y) + x)) && \text{by (assoc+)} \end{aligned}$$

**End of Proof of Lemma 5**

## Commutativity of $\_ \ast \_$

Lemma 4 [Right zero of  $\_ \ast \_$  (rz\*)]  $X \ast 0 = 0$

Proof of Lemma 4 By structural induction on  $X$ .

I. Base case

```
open PNAT2 .
-- check
red 0 * 0 = 0 .
close
```

II. Induction case

```
open PNAT2 .
-- fresh constants
op x : -> PNat .
-- IH
eq x * 0 = 0 .
-- check
red s(x) * 0 = 0 .
close
```

End of Proof of Lemma 4

## Commutativity of $\_ \ast \_$

Lemma 5 [Right successor of  $\_ \ast \_$  (rs\*)]  $X \ast s(Y) = (X \ast Y) + X$

Proof of Lemma 5 By structural induction on  $X$ .

I. Base case

```
open PNAT2 .
-- fresh constants
op y : -> PNat .
-- check
red 0 * s(y) = (0 * y) + 0 .
close
```

II. Induction case

```
open PNAT2 .
-- fresh constants
ops x y : -> PNat .
-- IH
eq x * s(Y) = (x * Y) + x .
-- check
red s(x) * s(y) = (s(x) * y) + s(x) .
close
```

End of Proof of Lemma 5



## Commutativity of $\_ \ast \_$

Theorem 4 [Commutativity of  $\_ \ast \_$  (assoc\*)]  $X \ast Y = Y \ast X$

Proof of Theorem 4 By structural induction on  $X$ .

I. Base case

```
open PNAT2 .
-- fresh constants
op y : -> PNat .
-- lemmas
eq X * 0 = 0 . -- (rz*)
-- check
red 0 * y = y * 0 .
close
```

II. Induction case

```
open PNAT2 .
-- fresh constants
ops x y : -> PNat .
-- lemmas
eq X * s(Y) = (X * Y) + X . -- (rs*)
-- IH
eq x * Y = Y * x .
-- check
red s(x) * y = y * s(x) .
close
```

End of Proof of Theorem 4

## Correctness of a Tail Recursive Factorial

Theorem 5 [Correctness of a Tail Recursive Factorial (trf)]

$\text{fact1}(X) = \text{fact2}(X)$

Proof of Theorem 5 By structural induction on  $X$ .

Let  $x$  be fresh constants of  $\text{PNat}$ .

I. Base case

What to show is  $\text{fact1}(0) = \text{fact2}(0)$ .

$\text{fact1}(0) \rightarrow s(0)$	by (f1-1)
$\text{fact2}(0) \rightarrow \text{sfact2}(0, s(0))$	by (f2)
$\rightarrow s(0)$	by (sf2-1)

## Correctness of a Tail Recursive Factorial

### II. Induction case

What to show is  $\text{fact1}(s(x)) = \text{fact2}(s(x))$

assuming the induction hypothesis  $\text{fact1}(x) = \text{fact2}(x) \quad \text{-- (IH)}$

$$\begin{aligned}
 \text{fact1}(s(x)) &\rightarrow s(x) * \text{fact1}(x) && \text{by (f1-2)} \\
 &\rightarrow s(x) * \text{fact2}(x) && \text{by (IH)} \\
 &\rightarrow s(x) * \text{sfact2}(x, s(0)) && \text{by (f2)} \\
 \text{fact2}(s(x)) &\rightarrow \text{sfact2}(s(x), s(0)) && \text{by (f2)} \\
 &\rightarrow \text{sfact2}(x, s(x) * s(0)) && \text{by (sf2-2)}
 \end{aligned}$$

We cannot make the two terms equal only by rewriting, and need a lemma.

The following is one candidate:

$$s(X) * \text{sfact2}(X, s(0)) = \text{sfact2}(X, s(X) * s(0))$$

This is so specific that the proof of this lemma needs another lemma.

Therefore, we make it more generic as follows:

$$Y * \text{sfact2}(X, Z) = \text{sfact2}(X, Y * Z) \quad \text{-- (p-sf2)}$$

$$\text{fact1}(s(x)) \rightarrow \text{sfact2}(x, s(x) * s(0)) \quad \text{by (p-sf2)}$$

End of Proof of Theorem 5

## Correctness of a Tail Recursive Factorial

**Lemma 6** [Property of sfact2 (p-sf2)]  $Y * \text{sfact2}(X, Z) = \text{sfact2}(X, Y * Z)$

**Proof of Lemma6** By structural induction on  $X$ .

Let  $x, y, z$  be fresh constants of  $\text{PNat}$ .

### I. Base case

What to show is  $y * \text{sfact2}(0, z) = \text{sfact2}(0, y * z)$ .

$$y * \text{sfact2}(0, z) \rightarrow y * z \quad \text{by (sf2-1)}$$

$$\text{sfact2}(0, y * z) \rightarrow y * z \quad \text{by (sf2-1)}$$

## Correctness of a Tail Recursive Factorial

### II. Induction case

What to show is  $y * \text{sfact2}(s(x), z) = \text{sfact2}(s(x), y * z)$

assuming the induction hypothesis

$Y * \text{sfact2}(x, Z) = \text{sfact2}(x, Y * Z) \quad \text{-- (IH)}$

$y * \text{sfact2}(s(x), z) \rightarrow y * \text{sfact2}(x, s(x) * z)$	by (sf2-2)
$\rightarrow \text{sfact2}(x, y * (s(x) * z))$	by (IH)
$\text{sfact2}(s(x), y * z) \rightarrow \text{sfact2}(x, s(x) * (y * z))$	by (sf2-2)
$\rightarrow \text{sfact2}(x, (s(x) * y) * z)$	by (assoc*)
$\rightarrow \text{sfact2}(x, (y * s(x)) * z)$	by (comm*)
$\rightarrow \text{sfact2}(x, y * (s(x) * z))$	by (assoc*)

End of Proof of Lemma 6

## Correctness of a Tail Recursive Factorial

Lemma 6 [Property of sfact2 (p-sf2)]  $Y * \text{sfact2}(X, Z) = \text{sfact2}(X, Y * Z)$

Proof of Lemma 6 By structural induction on  $X$ .


### I. Base case

```
open PNAT3 .
-- fresh constants
ops y z : -> PNat .
-- check
red y * sfact2(0, z)
    = sfact2(0, y * z) .
close
```

### II. Induction case

```
open PNAT3 .
-- fresh constants
ops x y z : -> PNat .
-- lemmas
eq (X + Y) * Z = (X * Z) + (Y * Z) . -- (d*o+)
-- IH
eq Y * sfact2(x, Z) = sfact2(x, Y * Z) .
-- check
red y * sfact2(s(x), z) = sfact2(s(x), y * z) .
close
```

Note that we need to use the lemma that was not used in the manual proof.



End of Proof of Lemma 6

Note that PNAT3 is PNAT2 in which `_*` is declared as follows:

**op** `_*` : PNat PNat -> PNat {**assoc comm**} .

## Correctness of a Tail Recursive Factorial

Theorem 5 [Correctness of a Tail Recursive Factorial (trf)]

$\text{fact1}(X) = \text{fact2}(X)$

Proof of Theorem 5 By structural induction on  $X$ .

I. Base case

```
open PNAT3 .
-- check
red fact1(0) = fact2(0) .
close
```

II. Induction case

```
open PNAT3 .
-- fresh constants
op x : -> PNat .
-- lemmas
eq Y * sfact2(X,Z) = sfact2(X,Y * Z) . -- (sf2-p)
-- IH
eq fact1(x) = fact2(x) .
-- check
red fact1(s(x)) = fact2(s(x)) .
close
```

End of Proof of Theorem 5

## Exercises

1. Write the specifications and proof scores used in the slides and feed them to the CafeOBJ systems.
2. Write two functions `sum1` and `sum2` that calculate the sum of natural numbers up to a given natural numbers (inclusive) that correspond to `fact1` and `fact2`, write manual proofs verifying that `sum1(X)` equals `sum2(X)` for all natural numbers  $X$ , and write proof scores formally verifying that `sum1(X)` equals `sum2(X)` for all natural numbers  $X$ .

## Appendices

The proof score of Theorem 1 is as follows:

"Theorem 1 [associativity of  $+$  (assoc+)]  
 $(X + Y) + Z = X + (Y + Z)$

Proof of Theorem 1. By structural induction on X.  
I. Base case"

```
open PNAT1 .  
  -- fresh constants  
  ops y z : -> PNat .  
  -- check  
  red (0 + y) + z = 0 + (y + z) .  
close
```

## Appendices

"II. Induction case"

```
open PNAT1 .  
  -- fresh constants  
  ops x y z : -> PNat .  
  -- IH  
  eq (x + Y) + Z = x + (Y + Z) .  
  -- check  
  red (s(x) + y) + z = s(x) + (y + z) .  
close
```

"End of Proof of Theorem 1"